

# **Оснoвы программирования на языке Си**

# Простейшая программа

---

главная (основная) программа  
всегда имеет имя *main*

```
main()  
{  
  
}  

```

начало  
программы

«тело»  
программы  
(основная  
часть)

конец  
программы



Что делает эта программа?

# Что происходит дальше?

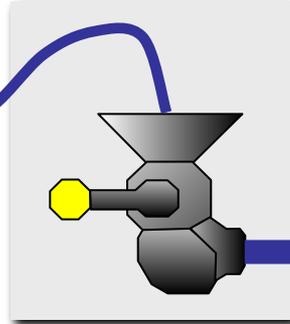
текст программы на Си или Си++

**first.cpp**

```
main ()
{
}
```

исходный файл

транслятор



**first.o**

```
ЪBzЦ2?|ё3БKa
n/36ШпlC+И-
Ц3_5MyPЧ6
s6bd^!/@:лЖ1_
```

объектный файл

стандартные  
функции



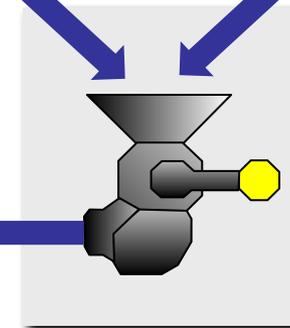
- по исходному файлу можно восстановить остальные
- исполняемый файл можно запустить

**first.exe**

```
MZPo:ЄPэ_e3"!_
`кп,ЦbЄ-Щр1
G_БАС,
_Ощяхя9жФ
```

исполняемый файл

редактор  
связей  
(компоновка)



# Вывод текста на экран

---

*include* = ВКЛЮЧИТЬ

```
#include <stdio.h>
main ()
{
printf ("Привет! ") ;
}
```

файл *stdio.h*:  
описание  
стандартных  
функций ввода  
и вывода

ВЫЗОВ СТАНДАРТНОЙ  
функции

*printf* = *print format*  
(форматный вывод)

ЭТОТ ТЕКСТ  
будет на  
экране



# Оболочка Dev C ++ 4.9

---

**IDE** = *Integrated Development Environment*

интегрированная среда разработки:

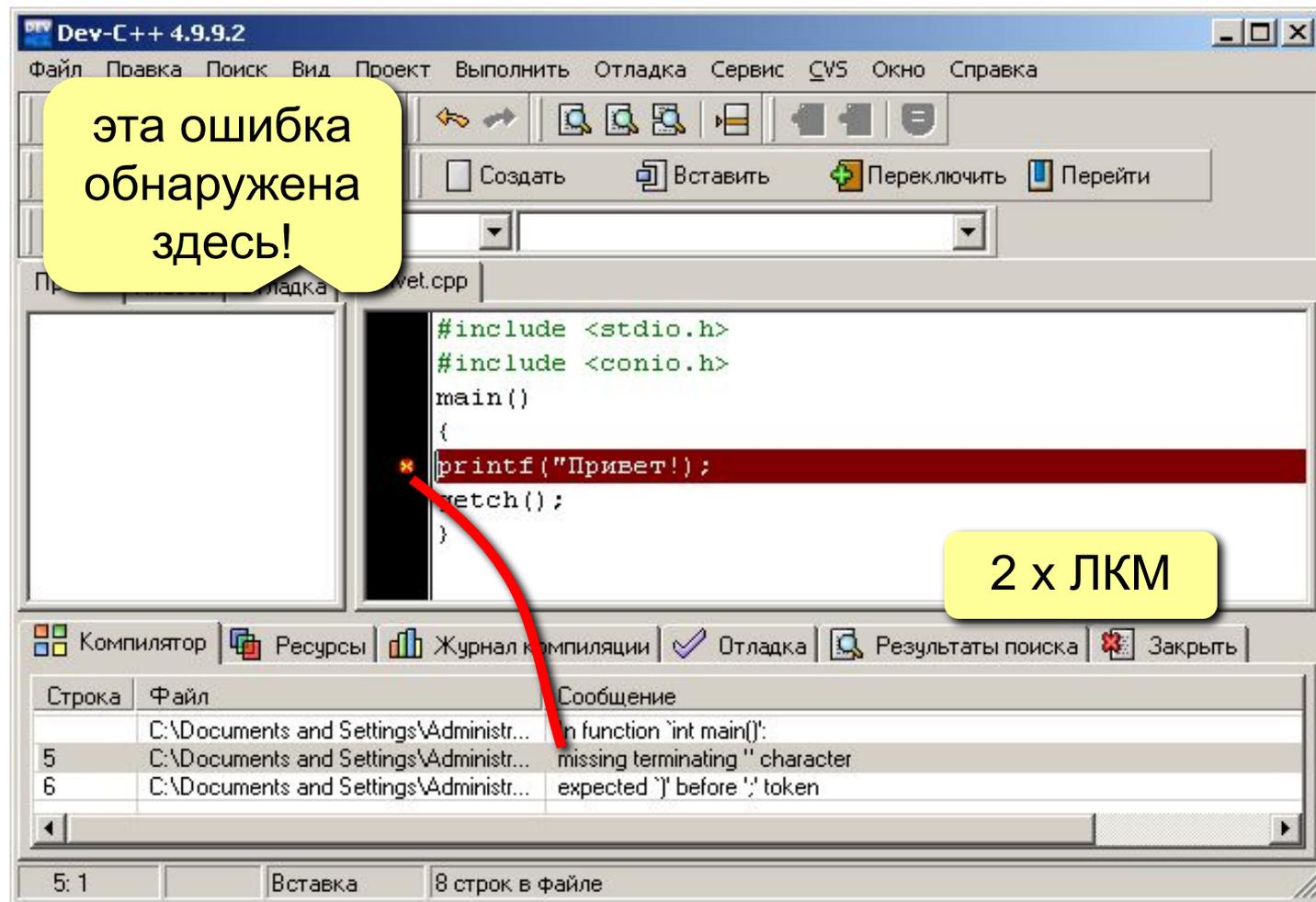
- **текстовый редактор** для создания и редактирования текстов программ
- **транслятор** для перевода текстов программ на Си и Си++ в команды процессора
- **КОМПОНОВЩИК** для создания исполняемого файла (EXE-файла), подключаются стандартные функции
- **отладчик** для поиска ошибок в программах

# Управление клавишами

---

Новый файл (Создать)	<b>Ctrl+N</b>	
Открыть файл	<b>Ctrl+O</b>	
Сохранить файл	<b>Ctrl+S</b>	
Закреть окно с программой	<b>Ctrl-F4</b>	
Запуск программы	<b>F9</b>	
Отменить	<b>Ctrl-Z</b>	
Восстановить отмененное	<b>Shift-Ctrl-Z</b>	

# Где ошибки?



**Ошибка может быть в конце предыдущей строки!**

# Наиболее «популярные» ошибки

---

**xxx.h: No such file or directory**

не найден заголовочный файл 'xxx.h' (неверно указано его имя, он удален или т.п.)

**'xxx' undeclared (first use this function)**

функция или переменная 'xxx' неизвестна

**missing terminating " character**

не закрыты кавычки "

**expected ;**

нет точки с запятой в конце оператора **в предыдущей строке**

**expected }**

не закрыта фигурная скобка

# Ждем нажатия любой клавиши

```
#include <stdio.h>
#include <conio.h>
main ()
{
printf ("Привет!"); // вывод на экран
getch (); /* ждать нажатия клавиши */
}
```

файл **conio.h**: описание функций для работы с клавиатурой и монитором

комментарий до конца строки

ждать нажатия на любую клавишу

комментарий между /\* и \*/

# Переход на новую строку

---

```
#include <stdio.h>
#include <conio.h>
main ()
{
printf ("Привет, \n Вася!");
getch ();
}
```

последовательность  
`\n` (код 10)  
переход на новую строку

на экране:

```
Привет ,
Вася!
```

# Задания

---

**«4»:** Вывести на экран текст "лесенкой"

Вася

пошел

гулять

**«5»:** Вывести на экран рисунок из букв

Ж

ЖЖЖ

ЖЖЖЖЖ

ЖЖЖЖЖЖЖ

НН НН

ZZZZZ

# Что такое переменная?

---

**Переменная** – это ячейка в памяти компьютера, которая имеет имя и хранит некоторое значение.

- Значение переменной может меняться во время выполнения программы.
- При записи в ячейку нового значения старое стирается.

## Типы переменных

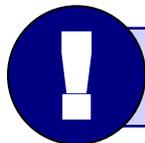
- **int** – целое число (4 байта)
- **float** – вещественное число, *floating point* (4 байта)
- **char** – символ, *character* (1 байт)

# Имена переменных

---

## Могут включать

- латинские буквы (A-Z, a-z)
- знак подчеркивания \_
- цифры 0-9



**Имя не может начинаться с цифры!**

## НЕ могут включать

- русские буквы
- пробелы
- скобки, знаки +, =, !, ? и др.

## Какие имена правильные?

**AXby R&B 4Wheel Вася "PesBarbos"**  
**TU154 [QuQu] \_ABBA A+B**

# Объявление переменных

**Объявить переменную** = определить ее имя, тип, начальное значение, и выделить ей место в памяти.

```
main ()
{
    int Tu104, I186=23, Yak42;
    float x=4.56, y, z;
    char c, c2='A', m;
}
```

целая переменная a

вещественные  
переменные

целые переменные  
**Tu104, I186 и Yak42**

вещественные  
переменные x, y и z  
**x = 4,56**

СИМВОЛЬНЫЕ  
переменные c, c2 и m  
**c2 = 'A'**

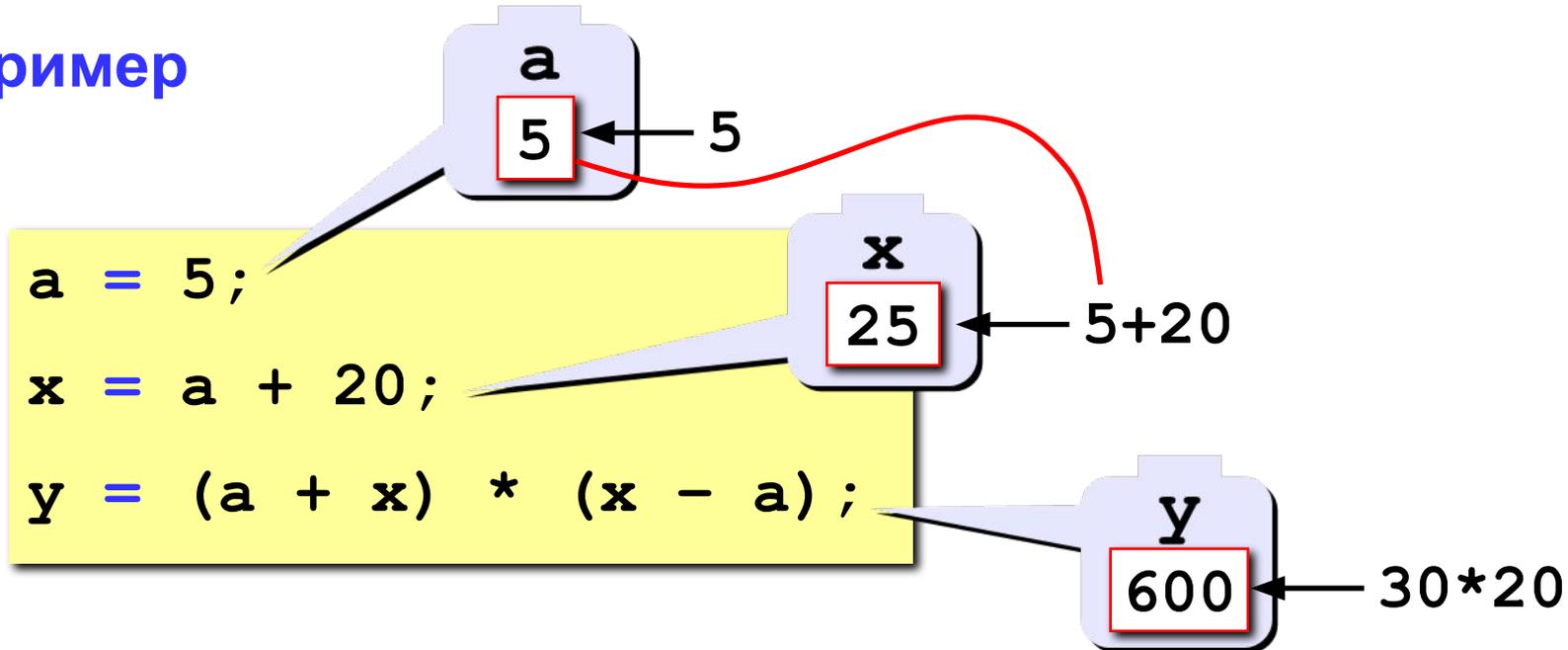
**!** Если начальное значение не задано, в этой ячейке находится «мусор»!

# Оператор присваивания

**Оператор** – это команда языка программирования высокого уровня.

**Оператор присваивания** служит для изменения значения переменной.

## Пример



# Оператор присваивания

Общая структура:

куда записать

что

*имя переменной = выражение;*

Арифметическое выражение может включать

- константы (постоянные)
- имена переменных
- знаки арифметических операций:

+ - \* / %

умножение

деление

остаток от  
деления

- вызовы функций
- круглые скобки ( )



Для чего служат  
круглые скобки?

# Какие операторы неправильные?

```
main ()
{
  int a, b;
  float x, y;
  a = 5;
  10 = x;
  y = 7,8;
  b = 2.5;
  x = 2*(a + y);
  a = b + x;
}
```

имя переменной  
должно быть слева  
от знака =

целая и дробная часть  
отделяются точкой

при записи вещественного  
значения в целую  
переменную **дробная**  
**часть будет отброшена**

# Особенность деления в Си

**!** При делении целых чисел остаток отбрасывается!

```
main ()
{
  int a = 7;
  float x;
  x = a / 4;
  x = 4 / a;
  x = float(a) / 4;
  x = 1.*a / 4;
}
```

1

0

1.75

1.75

# Сокращенная запись операций в Си

полная запись	сокращенная запись
<code>a = a + 1;</code> <span>инкремент</span>	<code>a++;</code>
<code>a = a + b;</code>	<code>a += b;</code>
<code>a = a - 1;</code> <span>декремент</span>	<code>a--;</code>
<code>a = a - b;</code>	<code>a -= b;</code>
<code>a = a * b;</code>	<code>a *= b;</code>
<code>a = a / b;</code>	<code>a /= b;</code>
<code>a = a % b;</code>	<code>a %= b;</code>

# Ручная прокрутка программы

```
main ()
{
    int a, b;
    a = 5;
    b = a + 2;
    a = (a + 2) * (b - 3);
    b = a / 5;
    a = a % b;
    a++;
    b = (a + 14) % 7;
}
```

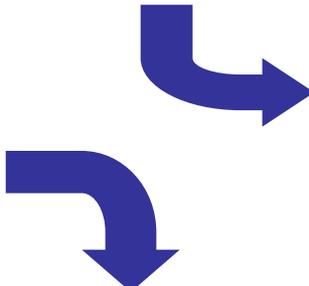
a	b
?	?
5	
	7
28	
	5
3	
4	
	4

# Порядок выполнения операций

- вычисление выражений в скобках
- умножение, деление, % слева направо
- сложение и вычитание слева направо

2 3 5 4 1 7 8 6 9

$$z = (5*a*c + 3*(c-d)) / a * (b-c) / b;$$

$$x = \frac{a^2 + 5c^2 - d(a+b)}{(c+d)(d-2a)}$$


$$z = \frac{5ac + 3(c-d)}{ab} (b-c)$$

2 6 3 4 7 5 1 12 8 11 10 9

$$x = (a*a + 5*c*c - d*(a+b)) / ((c+d) * (d-2*a));$$

# Сложение двух чисел

**Задача.** Ввести два целых числа и вывести на экран их сумму.

**Простейшее решение:**

```
#include <stdio.h>
#include <conio.h>
main ()
{
    int a, b, c;
    printf("Введите два целых числа\n");
    scanf ("%d%d", &a, &b);
    c = a + b;
    printf("%d", c);
    getch();
}
```

подсказка для  
ввода

ВВОД ДВУХ  
чисел с  
клавиатуры

Вывод результата

# Ввод чисел с клавиатуры

**scanf** –  
форматный ввод

формат ввода

адреса ячеек, куда  
записать введенные  
числа

```
scanf ("%d%d", &a, &b);
```

**Формат** – символьная строка, которая показывает, какие числа вводятся (выводятся).

**%d** – целое число

**%f** – вещественное число

**%c** – 1 символ

**%s** – символьная строка

**&a** – адрес  
переменной **a**

ждать ввода с клавиатуры двух целых чисел (через пробел или *Enter*), первое из них записать в переменную **a**, второе – в **b**

7652

12

**a** – значение  
переменной **a**

# Что неправильно?

```
int a, b;
```

```
scanf ("%d", a);
```

```
scanf ("%d", &a, &b);
```

```
scanf ("%d%d", &a);
```

**убрать пробел**

```
scanf ("%d %d", &a, &b);
```

```
scanf ("%f%f", &a, &b);
```

&a

%d%d

&a, &b

%d%d

# Вывод чисел на экран

здесь вывести  
целое число

это число взять  
из ячейки **c**

```
printf ("%d", c);
```

```
printf ("Результат: %d", c);
```

```
printf ("%d+%d=%d", a, b, c);
```

формат вывода

список значений

```
printf ("%d+%d=%d", a, b, a+b);
```

арифметическое  
выражение

# Вывод целых чисел

```
int x = 1234;  
printf ("%d", x);
```

1234

или "%i"

МИНИМАЛЬНОЕ  
ЧИСЛО ПОЗИЦИЙ

или "%9i"

```
printf ("%9d", x);
```

1234

всего 9 позиций

5

4

# Вывод вещественных чисел

```
float x = 123.4567;
printf ("%f", x);
```

123.456700

минимальное число  
позиций, **6 цифр** в  
дробной части

```
printf ("%9.3f",  
x);
```

123.456

всего 9 позиций,  
**3 цифры** в дробной  
части

```
printf ("%e", x);
```

1.234560e+02

стандартный вид:  
**1.23456·10<sup>2</sup>**

```
printf ("%10.2e", x);
```

1.23e+02

всего 10 позиций,  
**2 цифры** в дробной  
части мантииссы

# Полное решение

```
#include <stdio.h>
#include <conio.h>
main()
{
    int a, b, c;
    printf("Введите два целых числа\n");
    scanf("%d%d", &a, &b);
    c = a + b;
    printf("%d+%d=%d", a, b, c);
    getch();
}
```

ЭТО ВЫВОДИТ  
КОМПЬЮТЕР

## Протокол:

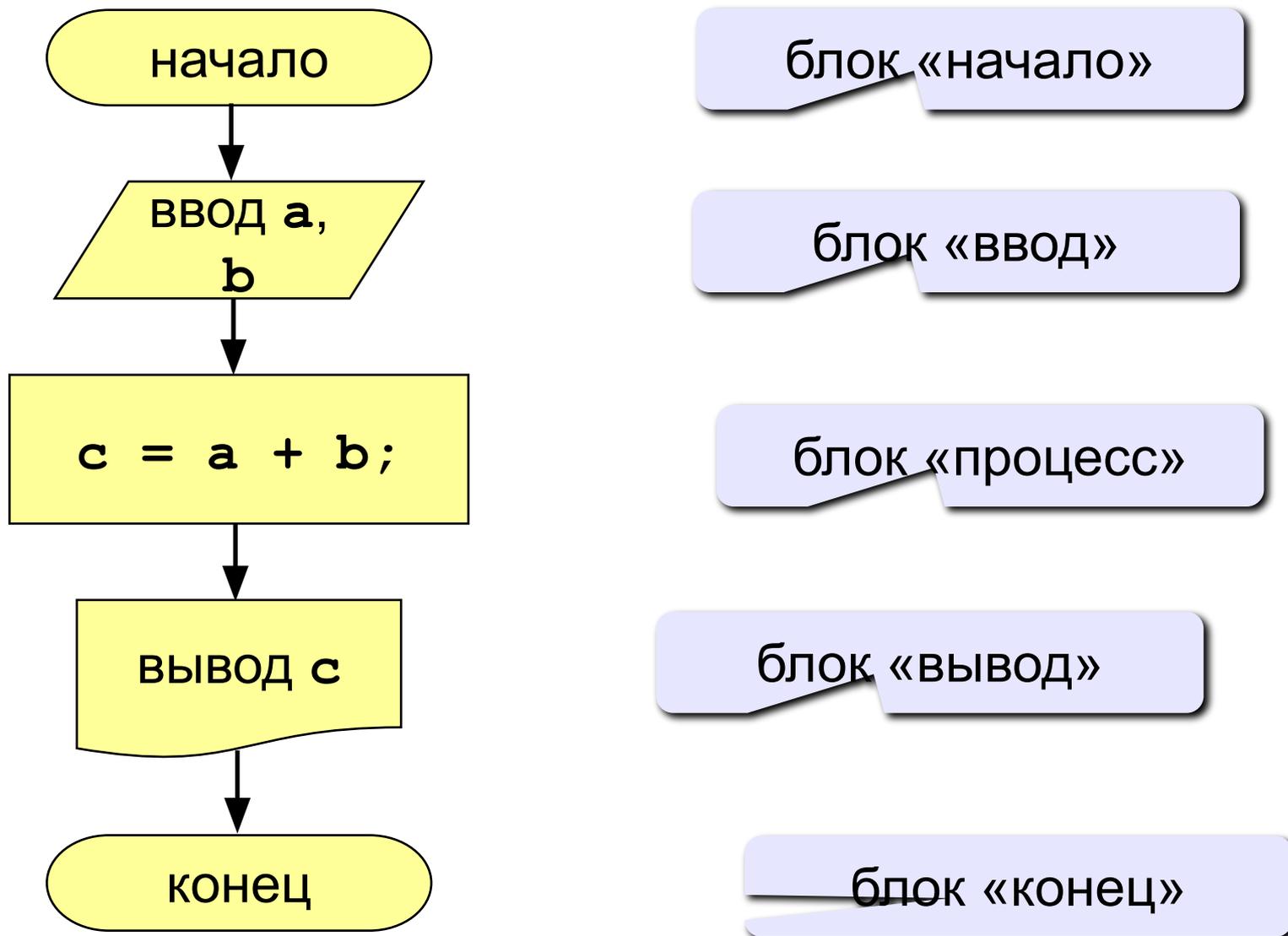
Введите два целых числа

25 30

25+30=55

ЭТО ВВОДИТ  
ПОЛЬЗОВАТЕЛЬ

# Блок-схема линейного алгоритма



# Задания

---

**«4»:** Ввести три числа, найти их сумму и произведение.

**Пример:**

Введите три числа:

4 5 7

$$4+5+7=16$$

$$4*5*7=140$$

**«5»:** Ввести три числа, найти их сумму, произведение и среднее арифметическое.

**Пример:**

Введите три числа:

4 5 7

$$4+5+7=16$$

$$4*5*7=140$$

$$(4+5+7) / 3 = 5.33$$

**СПАСИБО ЗА ВНИМАНИЕ!**